PRESIDENCY UNIVERSITY
KOLKATA

PRESIDENCY UNIVERSITY

Department of Mathematics

Ph.D. Thesis Synopsis

Title of the Proposed Thesis:

Mathematical Aspects of Cryptanalytic Attacks on Symmetric Key Cryptographic Algorithms

Name of the Candidate:	**Himadry Sekhar Roy**
Ph.D. Registration No.:	**R-21RS213120342**
Date of Ph.D. Registration:	**05.07.2022**
Name of the Supervisor:	**Prof. Avishek Adhikari**

Name of the Candidate:	**Himadry Sekhar Roy**
Name of the Supervisor:	**Prof. Avishek Adhikari**
Title of the Thesis:	**Mathematical Aspects of Cryptanalytic Attacks on Symmetric Key Cryptographic Algorithms**
Ph.D. Registration No.:	**R-21RS213120342**
Date of Ph.D. Registration:	**05.07.2022**
Place of Work:	**Presidency University, Kolkata**
Residencial Address:	**Vill Sardarhati** **PO Shason, PS Shason** **Dist: North24Parganas** **PIN: 743423, West Bengal, India**
Corresponding Address:	**Vill Sardarhati** **PO Shason, PS Shason** **Dist: North24Parganas** **PIN: 743423, West Bengal, India**
E-mail Id.:	**himadryddmc@gmail.com**
Contact No.:	**(+91) 8900669600**

Himadry Sekhar Roy
16.01.2026

Signature of the Research Scholar with Date

Avishek Adhikari

Signature of the Supervisor with Date

16/01/2026

Abstract

This thesis investigates the mathematical foundations of advanced cryptanalytic techniques on symmetric key cryptographic algorithms. It focuses primarily on block ciphers in both classical and emerging quantum settings. The work analyzes how structural properties of modern cipher designs, particularly lightweight and low-latency constructions, can be exploited using efficient key-recovery techniques beyond exhaustive search.

The thesis provides an overview of the historical development and foundational principles of cryptology, leading to modern cryptographic practice. After developing a rigorous theoretical framework for cryptanalysis, the thesis concentrates on single-key attacks, with emphasis on meet-in-the-middle (MITM) and biclique-based techniques. These methods bridge the gap between brute-force key search and differential-type attacks. A detailed cryptanalytic study is presented for the low-latency lightweight block ciphers `PRINCEv2` and `FUTURE` along with the prominent block cipher `AES`.

For `PRINCEv2`, the thesis presents effective key-recovery attacks based on a single biclique structure and a hybrid biclique with an improved Sieve-in-the-middle (SIM) based technique. The analysis undermines the prior security assertions made by the designers of `PRINCEv2`. The analysis shows how low-latency design choices introduce exploitable inter-round dependencies. For `FUTURE`, multiple overlapping biclique constructions are proposed, leading to a significant reduction in computational complexity. The analysis demonstrates that a linear key schedule in a block cipher facilitates the attainment of the maximum length biclique. In the context of `AES-128`, the development of advanced biclique and super S-box techniques has been undertaken to enhance time complexity, thereby surpassing the decade-long record in this field. For reduced 7-round `AES-128`, the MITM attack is applied utilising a known single plaintext-ciphertext pair. This includes refined attack techniques of recent advancements by reducing both data and memory requirements.

Beyond classical cryptanalysis, it formalizes the Quantum Biclique Attack (QBA) framework. Using Quantum Amplitude Amplification, classical biclique methods are extended to the quantum setting and applied to the block cipher `FUTURE`. A low-depth reversible quantum circuit for `FUTURE` is designed, together with detailed resource and gate-count estimates.

Overall, the results either overturn existing security claims or introduce new cryptanalytic techniques in both classical and quantum settings. They demonstrate that structural symmetries in lightweight cipher designs remain exploitable, even in the presence of targeted countermeasures.

Contents

1 Introduction to Cryptography **1**
- 1.1 Concise History of Cryptography . . . 1
- 1.2 Modern Cryptography . . . 2
- 1.3 Organisation of the Thesis . . . 3

2 Symmetric Key Cryptography **5**
- 2.1 Stream Ciphers . . . 5
- 2.2 Block Ciphers . . . 7
- 2.3 Cryptanalysis: An Overview . . . 8
  - 2.3.1 Security Notions . . . 8
  - 2.3.2 Adversarial Models . . . 8
  - 2.3.3 Complexity of a Cryptanalytic Technique . . . 9
  - 2.3.4 Various Type of Cryptanalytic Techniques . . . 10
- 2.4 A Brief Discussion on Some SPN based Block Ciphers . . . 10
  - 2.4.1 A Conceptual Description of `AES` . . . 11
  - 2.4.2 Description of `AES-128` . . . 11
- 2.5 Description of `PRINCEv2` . . . 12
- 2.6 Description of `FUTURE` . . . 14

3 Cryptanalysis of `PRINCEv2` Using Biclique Structures **16**
  - 3.0.1 Construction of a Biclique . . . 16
  - 3.0.2 Overview of the Attack Process . . . 17
- 3.1 Significant Results . . . 18

4 Cryptanalysis of Full Round FUTURE with Multiple Biclique Structures **20**
- 4.1 Proposed Biclique Structures in FUTURE . . . 20
  - 4.1.1 Formation of Biclique from Differentials . . . 21
- 4.2 Proposed Single Biclique Analysis of FUTURE . . . 22
  - 4.2.1 Biclique Analysis . . . 22
  - 4.2.2 Matching with Precomputation . . . 22
  - 4.2.3 Complexity of the Attack . . . 22
- 4.3 Significant Results . . . 23

5 Enhanced Biclique and MITM Based Key Recovery Attacks on Full and Reduced Round AES-128 **24**
5.1 Overall Attack Procedure 25
5.2 Significant Results 25

6 Low-Depth Quantum Implementation and Quantum Biclique Cryptanalysis of FUTURE **26**
6.1 Quantum Biclique Attack: a General Concept 26
6.2 Significant Results 27

7 Summary and Future Directions **28**

Bibliography **35**

Chapter 1

Introduction to Cryptography

We now explore cryptography more concisely for the readers. We begin by explaining its historical evolution, aiming to make the subject accessible to non-specialists, and conclude with an overview of its modern advancements.

1.1 Concise History of Cryptography

In cryptology, cryptography is one part, while cryptanalysis is the other. Cryptology originates from two Greek words, "kryptós" and "lógos", which imply "hidden or secret" and "study, or science" respectively, and is generally the science of constructing or analyzing secret codes.

In India, the concept of cryptography was first introduced in the Arthashastra, written by Kautilya, for secure communication. Arab mathematician Al-Kindi introduced a cryptanalytic technique for mono-alphabetic ciphers utilizing frequency analysis of a language around 800 AD.

At the end of 19^{th} century, the theoretical underpinnings of cryptography made significant strides, ushering in more systematic and constructive methods for its development. A fundamental principle, called Kerckhoff's principle, which serves as the bedrock for the development of modern cryptographic algorithms. This principle was first articulated by Auguste Kerckhoffs, a Dutch linguist and cryptographer, in 1883. In *La Cryptographie Militaire* [1], Kerckhoffs introduced six critical requirements for the construction of military cryptosystems. Among these, the most enduring and widely recognized principle asserts that the security of a cryptographic system should rely only on the secrecy of the key, not on keeping the algorithm a secret.

Rigorous research at the intersection of mathematics and computer science, which forms the groundwork for modern cryptographic development, began to emerge prominently during the World Wars. The transformation driven by the urgent need to secure communications and decrypt enemy codes. During the First World War, the German military employed advanced substitution ciphers to ensure secure communication, marking a significant evolution in the application and study of cryptographic techniques. Subsequently,

this cipher was successfully deciphered by a team of British cryptanalysts.

1.2 Modern Cryptography

Claude Shannon, the American cryptographer, was a leading contributor to the development of modern cryptographic algorithms. His first major work, entitled "A Mathematical Theory of Communication", published in 1948, laid the foundations of information theory and introduced key concepts such as information entropy, channel capacity, redundancy, and the bit as a unit of information. In 1949, he published another significant work entitled "Communication Theory of Secrecy Systems". The thesis is acknowledged as a seminal contribution to the development of modern cryptography. He introduced some key concepts such as perfect secrecy (demonstrating the theoretical unbreakability of the one-time pad), unicity distance, and the principles of confusion and diffusion. This work marked the shift from classical to modern cryptography by establishing a rigorous scientific principles essential for the design of secure ciphers, and it profoundly influenced subsequent advancements in the field.

The principles of confusion and diffusion in cryptographic algorithm development are as follows:

- Confusion- obscuring the mapping between plaintext and ciphertext.
- Diffusion- Spreading the effects of the plaintext influence across the ciphertext.

Starting in the early 1970s, the U.S. National Bureau of Standards (NBS, which later became National Institute of Standards and Technology (NIST) [2]) required a cryptographic algorithm for secure communication. At that time, Lucifer, developed by Horst Feistel, was later modified and introduced as DES [3] by IBM in 1975, in collaboration with the National Security Agency (NSA). In 1977, National Institute of Standards and Technology (NIST) standardized DES for encryption . It encrypts a 64-bit plaintext to a 64-bit ciphertext using a 56-bit secret key. DES remained influential for several decades.

Throughout these developments, cryptanalysis has played an indispensable role in identifying weaknesses. This process aims either to retrieve the secret (the secret key) or to distinguish its output from random, thereby compromising its security.

From a cryptanalysis perspective of symmetric ciphers, brute-force attacks are often the first consideration. Closely related, the meet-in-the-middle (MITM) attack was introduced by Diffie and Hellman in 1977, primarily demonstrating the vulnerability of Double DES (a successor to DES) [4].

A significant breakthrough in cryptanalysis came in 1990 when Eli Biham and Adi Shamir first introduced differential cryptanalysis [5], a powerful distinguishing technique they successfully implemented against DES. Their work was later published in 1993, marking a remarkable development in the field. Subsequently, in 1993, Mitsuru Matsui developed another potent technique known as linear cryptanalysis [6], which was also used to examine DES.

Growing security concerns regarding the Data Encryption Standard (DES) prompted NIST (National Institute of Standards and Technology) to organize a competition for the Advanced Encryption Standard (AES) in 1997. Rijndael [7], developed by two Belgian cryptographers Vincent Rijmen and Joan Daemen, was selected as the AES algorithm [8]. NIST officially standardized it for data encryption in 2001.

The creation of cryptographic primitives is an ongoing process, often fueled by rigorous research and numerous competitions. Notable examples include the eSTREAM competition (2004) for stream ciphers, the CAESAR competition (2012) for authenticated ciphers, and the NIST Lightweight Cryptography (LWC) competition (2019) for authenticated encryption with associated data.

Rigorous research and these competitions have yielded significant advancements in symmetric-key cryptography, leading to the development of ciphers such as:

- Stream Ciphers: Grain v1 [9], MICKEY 2.0 [10], Trivium [11], and Salsa20 [12] etc.
- Authenticated Ciphers: ACORN [13], AEGIS [14], Deoxys-II [15], and ASCON [16] etc.
- Lightweight Block Ciphers: AES [8], GIFT [17], Hight, PRESENT [18], Midori, PRINCE [19], and PRINCEv2 [20] etc.
- Authenticated Encryption with Associated Data (AEAD): ASCON AEAD [21]
- Cipher Structure based on Sum of PRFs: Orthros [22], Gleeok [23]

With the advent of the quantum era, the primary quantum threat to symmetric-key cryptography stems from Grover's algorithm, which offers a quadratic speedup for searching unsorted databases. In the context of cryptography, this means it can speed up a brute-force key search for symmetric ciphers. Fortunately, the impact of Grover's algorithm on symmetric-key cryptography is generally considered manageable. To maintain an equivalent level of security against a quantum adversary, one simply needs to double the key length.

1.3 Organisation of the Thesis

Chapter 2 provides a detailed overview of the structures of different symmetric key primitives, such as block ciphers and stream ciphers, and explores their development and operational principles using prominent examples like `PRINCEv2` and `FUTURE`. It also explores the historical context and design evolution of `AES`, which became the standard for global use by NIST, offering a comprehensive look at its structure and security properties. Furthermore, the chapter is dedicated to the different cryptanalytic techniques such as differential cryptanalysis, linear cryptanalysis, integral attack, MITM attack, biclique attack, and other generic attacks for symmetric key ciphers. A major application of this cryptanalytic work involves the analysis of fundamental cryptographic primitives, specifically block ciphers.

Chapter 3 introduces a biclique-based attack on a low-latency cipher PRINCEv2. Despite its reinforced key schedule and α-reflection middle layer, two independent full-round single-key attacks are mounted. One is a pure biclique attack with a time complexity of $2^{125.95}$ and the other is a hybrid technique, biclique with Sieve-in-the-middle (SIM) based technique with a time complexity of $2^{125.66}$. Both techniques have a maximum data and memory complexities of 2^8 chosen plaintexts and 2^{64} respectively. A reduced-round biclique on 8-round PRINCEv2 further validates the approach. These results overturn prior security claims and underscore that structural symmetries in lightweight designs remain exploitable even after targeted countermeasures. This chapter is based on the article titled "Cryptanalysis of PRINCEv2 using Biclique Structures", which was published in the journal "Discrete Applied Mathematics" in 2026 [24].

Chapter 4 presents cryptanalysis of full round FUTURE under a multi-biclique structure. A ciphertext side biclique spanning from round 6 to the final round delivers a computational complexity of $2^{125.8875}$ with data complexity upper bounded by 2^{48} and memory complexity of 2^{32}. By appending a second biclique between rounds 2 and 6 by introducing sub-partitioning of the keyspace, the computational complexity drops to $2^{125.5365}$. This chapter is based on the article titled "Cryptanalysis of Full Round FUTURE with Multiple Biclique Structures", which was published in the journal "Peer-to-Peer Networking and Applications" in 2024 [25].

Chapter 5 presents an advanced biclique-based technique applicable to both AES-128 and its reduced-round versions, achieving time complexity improvements with data complexities lower than searching the entire codebook. A dual-perspective biclique of dimension $2^{24} \times 2^{16}$ is constructed across rounds 8–10, paired with an enhanced SIM oracle that exploits byte-wise partial-matching. The resulting full-round AES-128 attack achieves $2^{125.60}$ time (memory 2^{64}, data 2^{104}). For reduced-round targets, the framework sets new records: 8-round AES-128 in $2^{122.29}$, 7-round AES-128 in $2^{120.63}$ via biclique, and an ultra-low-data MITM variant on 7 rounds requiring one known plaintext–ciphertext pair yet recovering the key in $2^{125.10}$ (with same data and memory 2^{64}). This chapter is based on the communicated article titled, "Enhanced Biclique and MITM Based Key Recovery Attacks on Full and Reduced Round AES-128".

Chapter 6 refers to Quantum Biclique Attack (QBA) on FUTURE along with a low-cost quantum implementation of the block cipher FUTURE. The classical biclique attack is adapted to the quantum setting through the application of Quantum Amplitude Amplification (QAA) to effectively address non-uniform entangled states. This adaptation successfully forges a direct theoretical and practical bridge between classical and quantum cryptanalysis methodologies. This chapter is based on the communicated article titled, "Low-Depth Quantum Implementation and Quantum Biclique Cryptanalysis of FUTURE".

The thesis concludes with the Chapter 7. This concluding chapter does not merely summarize the findings presented earlier; rather, it deliberately pivots toward the open frontiers of the field, offering a forward-looking roadmap for the cryptanalysis of symmetric-key cryptographic primitives.

Chapter 2

Symmetric Key Cryptography

In symmetric key cryptography, a significant paradigm in the field of cryptography, both encryption and decryption rely on a single secret key, also referred to as secret key cryptography. Prior to the communication, the symmetric key, denoted K, is securely shared between Alice and Bob.

Symmetric key cryptography is fundamentally divided into two distinct paradigms: stream ciphers and block ciphers. Details of the components of secure communications are outlined below.

- **Confidentiality:** To ensure that the information of a communication is accessible only to the authorized parties. The encryption mechanism of a cryptosystem ensures that the decryption of ciphertext is computationally infeasible without possession of the secret key.
- **Integrity:** To ensure that the data has not been altered, destroyed, or tampered with by an eavesdropper during communication. Solving the problem related to integrity, a tag is generated along with the ciphertext of a cryptosystem. Consequently, an eavesdropper can not generate a valid tag and ciphertext pair without possessing the secret key.

In modern cryptography, combining confidentiality and integrity introduces authenticated encryption. Recently, ASCON was named a finalist (2019) in the NIST competition for lightweight authenticated encryption with associated data.

2.1 Stream Ciphers

The development of stream ciphers is rooted in the principles of the One-Time Pad (OTP), which was first described by Gilbert S. Vernam in 1917 [26], building on his earlier work with teletype machines. The encryption mechanism involves XORing the plaintext string with a random key stream of the same length to produce the ciphertext. In 1949, Claude E. Shannon formally proved the perfect secrecy of the OTP.

To ensure perfect secrecy, the One-Time Pad requires the secret master key to be as long as the plaintext and uniformly random. Recognizing the impracticality of this in many scenarios, cryptographers developed stream ciphers to generate a pseudo-random key stream of arbitrary length from a much shorter, fixed-length master key (most likely 80, 128, 192, or 256 bits). Also in the realm of stream ciphers, the encryption function is a straightforward operation: the bitwise XOR of the plaintext with the generated keystream. The initial approach to key stream generation in stream ciphers, emerging in the 1960s, was the use of Linear Feedback Shift Registers (LFSRs). The cryptographic complexity and security of these ciphers do not reside in the encryption process itself but rather in the sophisticated and non-linear mechanisms often employed along with LFSRs to introduce non-linearity and produce a highly unpredictable, pseudo-random keystream.

The GSM A5/1 [27], a canonical example of a stream cipher, has a design based entirely on LFSRs. Information about the design of GSM A5/1 was not secret in 1994, and the algorithm of A5/1 was later revealed by Briceno *et. al.* in 1999 using reverse engineering.

Apart from designs based on LFSRs, which were susceptible to the correlation attack and the Berlekamp-Massey algorithm, cryptographers developed nonlinear feedback shift registers (NFSRs), i.e., feedback shift registers (FSRs) with a nonlinear update function. The development of NFSR-based stream ciphers started in the late 20^{th} century. One of the most popular NFSR-based stream ciphers is RC4, which was designed for RSA security (mainly data security) by an American computer scientist, Ron Rivest, in 1987. The concept of NFSR-based stream ciphers continues to be developed, with recent examples of stream ciphers including the Grain family, Lizard, Plantlet, and Espresso.

After vulnerabilities were found in RC4 and A5/1 using cryptanalytic techniques such as [28, 29, 30, 31, 32, 33, 34]. The evolution of stream ciphers has been substantially advanced through the contributions of major competitions such as the ECRYPT eSTREAM (2004-2008), CAESAR (2013-2019), and NIST LWC (2019-2023). These cryptographic competitions have catalyzed significant advancements in the development of stream ciphers and their derivative constructs, ensuring their suitability for the rigorous demands of modern cryptographic applications within high-speed, insecure network environments. The development of Salsa20, Grain, and Trivium was significantly shaped by the foundational ECRYPT eSTREAM project. In contrast, the CAESAR competition expanded the scope to authenticated encryption with associated data (AEAD), producing versatile designs such as Ascon and AEGIS. More recently, the NIST LWC competition has driven advancements in standardized lightweight cryptography, with Ascon emerging as a flagship outcome. The subsequent section delineates the methodology and outcomes of the three major antecedent competitions, as detailed below:

- **The eSTREAM Project (2004–2008):** The eSTREAM project, an initiative of the European Network of Excellence for Cryptology (ECRYPT), was conducted from 2004 to 2008 to identify secure and efficient stream ciphers for widespread use. The project developed a portfolio of stream ciphers suitable for both software-oriented applications (with an emphasis on high-throughput) and hardware-oriented implementations. The successful completion of a three-phase evaluation, involving ex-

tensive cryptanalysis, the key outcomes identified for software were HC-128, Rabbit, Salsa20, and SOSEMANUK, while the ciphers selected for hardware were F-FCSR-H, F-FCSR-16, Grain, Trivium, and MICKEY.

- **The CAESAR Competition (2013–2019):** In 2013, the CAESAR (Competition for Authenticated Encryption: Security, Applicability, and Robustness) competition was initiated with a focus on AEAD schemes. After successful evaluation, involving extensive cryptanalysis, the key outcomes identified for lightweight categories ware ACORN, Ascon, while the ciphers selected for high-performance and in depth applications ware AEGIS, MORUS, OCB Deoxys-II, COLM.
- **The NIST LWC Competition (2019–2023):** The Lightweight Cryptography (LWC) competition was launched by NIST in 2019 with the goal of standardizing algorithms suitable for lightweight environments. This competition specifically focused on both AEAD and hash function schemes. Out of an initial submission of 56 candidates, only eight were selected for the final round. Subsequently, ASCON was chosen as the selected algorithm for the NIST LWC competition.

Current research trends in stream cipher development are centered on two primary directions, one is sponge and duplex based constructions notably ASCON [21], Elephant [35], Xoodyak [36], and other prominent AEAD ciphers. Another direction of research is the design construction and security analysis of fully homomorphic encryption (FHE)-based stream ciphers like Rasta [37], Masta [38] etc.

2.2 Block Ciphers

Another highly prominent and fundamental symmetric-key primitive is the block cipher. A block cipher encrypts a fixed-size plaintext block into a ciphertext block of the same size, utilizing a secret key block which is often, but not always, larger than the plaintext block. The encryption process is commonly structured as a repetition of identical transformation, which is known as round function. The round function is the main primitive of the block ciphers. Block ciphers constructed in this manner are often referred to as iterative block ciphers. If a block cipher, E encrypt n-bit plaintext to n-bit ciphertext using k-bit secret key then mathematically it can be expressed as:

$$E : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n.$$

Now a block cipher for a particular secret key $K \in \{0,1\}^k$ is a bijective function from $\{0,1\}^n$ to itself and defined as:

$$E_K : \{0,1\}^n \to \{0,1\}^n.$$

If E_K^{-1} is the inverse of E_K and P be a member of the plaintext space, $\{0,1\}^n$ then it satisfy the correctness principal as:

$$P = E_K^{-1}\left(E_K\left(P\right)\right).$$

Block cipher constructions are fundamentally based on the principles of confusion and diffusion. The concept first formally proposed by the American computer scientist Claude Shannon in his one of the most influential 1949 work, "Communication Theory of Secrecy Systems".

The subsequent section will explore the internal architectures of several prominent block cipher constructions. Block ciphers are primarily categorized into three fundamental design architecture:

- The Feistel cipher
- The Substitution-Permutation Network (SP-Network or SPN)
- The ARX structure (Addition-Rotation-XOR)

2.3 Cryptanalysis: An Overview

Symmetric-key primitives encompass diverse cryptographic structures, primarily stream ciphers and block ciphers. Due to these structural differences, various cryptanalytic techniques are employed and implemented to analyze the security of each cipher type and ultimately to break the cipher and retrieve the secret key. Various cryptanalytic techniques not only validate existing algorithms but also open new directions for developing future schemes. The section will therefore focus on several basic and well-established cryptanalytic methods applied to symmetric key primitives. Since my thesis is based on the cryptanalysis of symmetric key primitives mostly based on some block ciphers, in the section, first discuss about the adversarial models along with security notation of symmetric key ciphers.

2.3.1 Security Notions

Different security notions are implemented to analysis a symmetric key primitive. The major one is **resistance of the secret key recovery** but for block ciphers, others are **plaintext recovery resistance**, and **indistinguishability form a random permutation**. If first one is broken for a block cipher, then other two are automatically broken. Breaking a cryptosystem, depends on which security notion is broken. From a cryptanalysis perspective, **a key recovery attack** is evaluated as the most significant achievement in this domain. For a cryptographer, the notion called **resistance of the secret key recovery** is considered as the weak security notion relative to the other two.

2.3.2 Adversarial Models

The analysis of any symmetric key algorithm depends on the adversarial model used. Adversarial attack models are categorized into four types, which will be detailed in the subsequent section.

- ***Ciphertext Only Attack (COA):*** In this type, the adversary is limited to accessing only the set of ciphertexts to mount the attack.

- ***Known Plaintext Attack (KPA):*** In this attack scenario, the adversary is restricted to access a set of known plaintext-ciphertext pairs to mount the attack.

- ***Chosen Plaintext Attack (CPA):*** This adversarial attack model permits the adversary to query a encryption oracle with chosen plaintexts and receive the corresponding ciphertexts under a fixed secret key, using this information to mount the attack.

- ***Chosen Ciphertext Attack (CCA):*** This adversarial attack model permits the adversary to query a decryption oracle with chosen ciphertexts and receive the corresponding plaintexts under a fixed secret key, using this information to mount the attack.

In cryptanalysis, the strength of an attack model is inversely proportional to the information and control required by the adversary. Specifically, an attack model is considered 'strong' if it yields the key while providing the adversary with minimal information. For instance, the ciphertext-only attack (COA) is considered the strongest model because the adversary requires only the ciphertext to succeed, indicating minimal necessary capability. Conversely, the chosen-ciphertext attack (CCA) is considered weaker as an attack model because it requires a powerful adversary who can actively interact with the decryption oracle, demonstrating maximal capability and system access.

In the context of cryptanalysis, an encryption or decryption oracle is defined as a mechanism that allows an attacker to query a cipher's encryption or decryption mechanism with the unknown secret key from plaintext or ciphertext to obtain corresponding ciphertext or plaintext.

2.3.3 Complexity of a Cryptanalytic Technique

Complexity measure is primarily associated with three distinct costs:

- **Time Complexity (T):** The cost of computational operations are needed to execute the attack or the cost during online phase of a attack. It is also called computational cost.

- **Memory complexity (M):** The amount of storage is required to execute the attack or the cost during offline phase of the attack.

- **Data Complexity (D):** The quantity of oracle queries needed to mount the attack.

2.3.4 Various Type of Cryptanalytic Techniques

Cryptanalysis is a rigorous process to analyze a symmetric key primitives. Development of cryptanalysis also has an fascinating history. This section discusses various cryptanalysis techniques applicable to block ciphers. Specifically the section will focus on the different type of cryptanalysis methods based on differential cryptanalysis.

Let E be a block cipher with total round r, maps n-bit plaintext block to n-bit ciphertext block using k-bit secret key. Similarly E such that:

$$E : \{0,1\}^n \times \{0,1\}^k \to \{0,1\}^n.$$

If $C = E_K(P)$ where $P, C \in \{0,1\}^n$ and $K \in \{0,1\}^k$, the C is the ciphertext of P with respect to the encryption function E under the secret key K.

For the block cipher E, it is a necessary condition that the key size k must be greater than or equal to the block size, n i.e., $(k \geq n)$. Then the minimum number of plaintext-ciphertext pairs is required to retrieve the secret key of the block cipher E is determined by $\lceil \frac{k}{n} \rceil$. Various types of cryptanalysis are given below:

- Brute-Force (Exhaustive Search) Attack
- Codebook Attack
- Time Memory Data Trade-off (TMDTO) Attack
- Differential Cryptanalysis
- Linear Cryptanalysis
- Integral Attack
- Differential Fault Attack (DFA)
- Meet-in-the-middle (MITM) Attack (1977)
- Biclique Attack (2011)

2.4 A Brief Discussion on Some SPN based Block Ciphers

Here several selected SPN structured block ciphers are explored. The section deals with a brief discussion on design and analysis of `AES`, `PRINCEv2`, and `FUTURE`. Initially, the section will focus on design of `AES`, followed by `PRINCEv2` and `FUTURE`. The analysis section will then briefly discuss about the existing attacks on these ciphers, with a primary emphasis on biclique-based cryptanalytic techniques and the MITM attack.

2.4.1 A Conceptual Description of AES

Advanced Encryption Standard (AES) [8] is a widely recognized substitution permutation network (SP-Network) based block cipher. In 1997, the National Institute of Standards and Technology (NIST) [2] initiated a competition to select the Advanced Encryption Standard (AES). The Rijndael algorithm [7], developed by cryptographers Vincent Rijmen and Joan Daemen, was submitted as a candidate, it was subsequently chosen as an AES.

Rijndael is a family of block ciphers designed to be flexible, supporting block sizes of 128, 160, 192, 224, and 256 bits. It also accommodates corresponding key sizes of 128, 160, 192, 224, and 256 bits. However, AES adopted a fixed 128-bit block size with key sizes of 128, 192, and 256 bits. AES [8] was officially standardized for data encryption in 2001.

2.4.2 Description of AES-128

AES is the most popular in the SPN-based block cipher categories. AES-128 operates on a 128-bit plaintext P and generates a 128-bit ciphertext C using a 128-bit master key K. The encryption process consists of 10 rounds. Initially, during the key whitening phase, the plaintext P is XORed with the initial round key $RK_0 = K$ to obtain the initial state $S^0 = P \oplus RK_0$.

Each of the subsequent 9 rounds consists of a composition of three functions: SubBytes (SB), ShiftRows (SR), and MixColumns (MC), followed by XOR with the round key RK_i. The transformation can be expressed as:

$$S^i = RF(S^{i-1}, RK_i) = RK_i \oplus MC \circ SR \circ SB(S^{i-1}),$$

where S^i is the 128-bit state after the i^{th} round for $i = 1, \ldots, 9$, and RF represents the round function.

In the final round, the output state S^9 of the 9^{th} round is processed by SB, SR, and XOR with the final round key RK_{10} to produce the ciphertext (C), described as:

$$C = S^{10} = RK_{10} \circ SR \circ SB(S^9).$$

Let $S^4_{\text{SB}} = SB(S^3)$ be the state of the cipher after the SubBytes (SB) operation at round 4. For notational convention, the first four bytes of S^4_{SB} will be denoted as $S^4_{\text{SB}}[0:3]$.

AES-128 can be expressed as a composition of three sub-procedures f, g, and h such that:

$$\text{AES-128} = h \circ g \circ f,$$

where $S^0 \xrightarrow[f]{K} S^2$, $S^2 \xrightarrow[g]{K} S^7$, and $S^7 \xrightarrow[h]{K} S^{10}$.

Let $\mathcal{K} = \{0,1\}^{128}$ be the keyspace of AES-128, with $|\mathcal{K}| = 2^{128}$, and let $\mathcal{K}_8 = \{0,1\}^{128}$ be the keyspace of AES-128 at round 8. The key-scheduling algorithm of AES-128 defines a bijection from $\mathcal{K}$ to $\mathcal{K}_8$.

Therefore, we denote $S^0 \xrightarrow[f]{RK} S^2$, $S^2 \xrightarrow[g]{RK} S^7$, and $S^7 \xrightarrow[h]{RK} S^{10}$ to imply the computations $S^0 \xrightarrow[f]{K} S^2$, $S^2 \xrightarrow[g]{K} S^7$, and $S^7 \xrightarrow[h]{K} S^{10}$, respectively, where RK represents a round key derived from the master key K for any chosen round. For comprehensive information regarding the key-scheduling algorithm of `AE-128`, refer to [8].
Note that, Chapter 5 relies on the description of `AES-128` provided in Section 2.4.2 to mount both cryptanalysis techniques, biclique and MITM.

2.5 Description of `PRINCEv2`

`PRINCEv2` is a lightweight, low-latency, AES-like SP-network (SPN)-based block cipher featuring a state size of 64 bits and a master key size of 128 bits. It was devised by Božilov et al. and introduced at Selected Areas in Cryptography (SAC), 2020 [20]. This cipher is an enhanced version of the `PRINCE` block cipher [19], incorporating modifications to the key schedule, round constants, and middle layer. `PRINCEv2` encrypts a 64-bit plaintext P into a 64-bit ciphertext C through 12 iterative rounds using a 128-bit master key K.

Each 64-bit state of `PRINCEv2` is depicted as 16 nibbles, with each nibble corresponding to a 4-bit state value. Now each round of `PRINCEv2` includes an S-box layer `SB`, a MixColumns operation `MC`, and a ShiftRows operation `SR`. In the S-box layer, a 4×4 S-box is applied uniformly across all 16 positions. The MixColumns operation uses an involutory matrix M' of size 64×64. The `SR` operation is identical to the ShiftRows operation used in a round of `AES` [8]. Since MixColumns operation of `PRINCEv2` is self invertible, $\texttt{SB}^{-1}$ and $\texttt{SR}^{-1}$ represent the inverse of `SB` and `SR` respectively.

In the key scheduling phase, the master key K is split up into two 64-bit parts K_0 and K_1 as $K = K_0||K_1$. RK_i symbolizes the round key of the i^{th} round. The round keys RK_0 and RK_{11} serve as whitening keys, applied via XOR at the beginning and end of the cipher.

The cipher is initialized as $S^0 = P \oplus K_0$. Let $\mathbf{R}_i$ denote the forward round function, defined as:

$$\mathbf{R}_i(S) = RC_i \oplus RK_i \oplus R(S),$$

where S is a state and $R = SR \circ MC \circ SB$. Here $\mathbf{R}_i^{-1}$ represents the corresponding backward round function.

Now the subsequent five rounds are updated by $S^{i+1} = \mathbf{R}_i(S^i) = R(S^i) \oplus K_{(i \mod 2)} \oplus RC_i$ for $i = 0, \ldots, 4$. This is followed by a middle layer transformation R' resulting in the state $S_m = R'(S^5)$ where

$$R'(S): \begin{cases} S \leftarrow K_0 \oplus SB(S) \\ S \leftarrow (RC_{11} \oplus K_1) \oplus MC(S) \\ S \leftarrow SB^{-1}(S) \end{cases}$$

for state value S. Next $S^6 = R^{-1}(S_m \oplus K_0 \oplus RC_6)$ and ensuing four rounds are updated as $S^{i+1} = R^{-1}\left(S^i \oplus K_{(i \mod 2)} \oplus RC_i\right)$ for $i = 6, \ldots, 9$ with the final output updated as

$$C = RC_{11} \oplus RK_{11} \oplus S^{10}.$$

In the decryption process, for ciphertext C, the master key is specified by $K = K_1 \oplus \beta || K_0 \oplus \alpha$ instead of $K = K_0 || K_1$, with RC_i for $i \in \{0, \ldots, 11\}$ and α, β are the input to the decryption algorithm. The algorithm returns the plaintext P. For a detailed explanation of the encryption and decryption process, refer to Algorithms 1 and 2 respectively.

One should note that to facilitate the proposed attack gracefully the following notations are used:

- $S^r[i]$ denotes the i^{th} nibble of the state S^r after the round r.
- $S^r[12:15] = (S^r[12], S^r[13], S^r[14], S^r[15])$
- $K_0[12:15] = (K_0[12], K_0[13], K_0[14], K_0[15])$

Algorithm 1: Encryption process of PRINCEv2.

Input: P, $K = K_0||K_1$, and RC_i for $i = 0, 1, 2, \ldots, 11$
Output: C

Initialization :
for $i = 0$ *to* 11 **do**
  $RK_i = K_{(i \mod 2)}$
end
$State \leftarrow P \oplus RK_0$
for $i = 1$ *to* 5 **do**
  $State \leftarrow R(State)$
  $State \leftarrow State \oplus RK_i \oplus RC_i$
end
$State \leftarrow SB(State)$
$State \leftarrow State \oplus K_0$
$State \leftarrow MC(State)$
$State \leftarrow State \oplus K_1 \oplus RC_{11}$
$State \leftarrow SB^{-1}(State)$
for $i = 6$ *to* 10 **do**
  $State \leftarrow State \oplus RK_i \oplus RC_i$
  $State \leftarrow R^{-1}(State)$
end
$C \leftarrow State \oplus RK_{11} \oplus RC_{11}$
return C

Algorithm 2: Decryption process of PRINCEv2.

Input: C, $K = K_0||K_1$, RC_i for $i = 0, 1, 2, \ldots, 11$, α, and β
Output: P

Initialization :
for $i = 0$ *to* 5 **do**
  if $i \mod 2 == 0$ **then**
    $RK_i = K_1 \oplus \beta$
  else
    $RK_i = K_0 \oplus \alpha$
  end
end
$State \leftarrow C \oplus RK_0$
for $i = 1$ *to* 5 **do**
  $State \leftarrow R(State)$
  $State \leftarrow State \oplus RK_i \oplus RC_i$
end
$State \leftarrow SB(State)$
$State \leftarrow State \oplus K_1 \oplus \beta$
$State \leftarrow MC(State)$
$State \leftarrow State \oplus K_0$
$State \leftarrow SB^{-1}(State)$
for $i = 6$ *to* 11 **do**
  if $i \mod 2 == 0$ **then**
    $RK_i = K_1 \oplus \alpha$
  else
    $RK_i = K_0 \oplus \beta$
  end
end
for $i = 6$ *to* 10 **do**
  $State \leftarrow State \oplus RK_i \oplus RC_i$
  $State \leftarrow R^{-1}(State)$
end
$P \leftarrow State \oplus RK_{11} \oplus RC_{11}$
return P

2.6 Description of FUTURE

FUTURE [39], an SPN-based iterated block cipher, produces 64-bit ciphertext C from 64-bit plaintext P using a 128-bit secret key K. The number of rounds of FUTURE is 10. The round function (RF) of FUTURE is the composition of SubCell (SC), MixColumn (MC), ShiftRow (SR) XORed with the round key RK, defined as $RF(s, RK) = RK \oplus SR \circ MC \circ SC(s)$, where s is a 64-bit block. Table 2.1 depicts 4-bit S-box, which is used in the

SubCell function of FUTURE.

Table 2.1: S-box of FUTURE.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$S(x)$	1	3	0	2	7	e	4	d	9	a	c	6	f	5	8	b

The 64-bit round key RK_i for the round i is generated using Algorithm 3. The Algorithm 4 represents the entire encryption process of FUTURE. Here $\oplus$ represents the bit-wise XOR operation and $\lll$ represents the left circular shift as described in [39].

Algorithm 3: KeySchedule of FUTURE.

Input: K and RC_i for $i = 1, 2, \ldots, 10$
Output: RK_i for $i = 0, 1, 2, \ldots, 10$

Initialization :
$X \leftarrow k_0, k_1, \ldots, k_{63}$
$Y \leftarrow k_{64}, k_{65}, \ldots, k_{127}$
for $i = 0$ *to* 10 **do**
  if $2 \mid i$ **then**
    $RK_i \leftarrow \left[X \lll \left(5 \times \frac{i}{2}\right)\right]$
    $RK_i \leftarrow RK_i \oplus RC_i$
  end
  if $2 \nmid i$ **then**
    $RK_i \leftarrow \left[Y \lll \left(5 \times \lfloor \frac{i}{2} \rfloor\right)\right]$
    $RK_i \leftarrow RK_i \oplus RC_i$
  end
end
return RK_i for $i = 0, 1, 2, \ldots, 10$

Algorithm 4: Encryption process of FUTURE.

Input: P and RK_i for $i = 0, 1, 2, \ldots, 10$
Output: C

Initialization :
$State \leftarrow P \oplus RK_0$
for $i = 1$ *to* 9 **do**
  $State \leftarrow RF(State, RK_i)$
end
$State \leftarrow SC(State)$
$State \leftarrow SR(State)$
$C \leftarrow State \oplus RK_{10}$
return C

Note that, Chapter 4 relies on the description of FUTURE provided in Section 2.6 to mount biclique based cryptanalysis technique.

Chapter 3

Cryptanalysis of PRINCEv2 Using Biclique Structures

This Chapter presents a comprehensive cryptanalysis of PRINCEv2, a lightweight block cipher optimized for low-latency applications. Using biclique cryptanalysis, we expose vulnerabilities in its design, particularly in its updated key schedule and middle layer operations. While PRINCEv2 was designed to resist biclique cryptanalysis, the work's findings indicate that such attacks remain feasible. We propose two novel single-key attacks on the full-round PRINCEv2, one based on biclique structures and the other improving upon it with a Sieve-in-the-Middle (SIM) based attack, achieving key recovery with time complexities of $2^{125.95}$ and $2^{125.66}$, respectively, with practical data and memory complexities. We further mounted a reduced-round biclique cryptanalysis on PRINCEv2. These findings provide critical insights into the cipher's security margins, challenging prior assumptions and underlining the need for addressing structural vulnerabilities in lightweight cryptographic algorithms.

3.0.1 Construction of a Biclique

We start with a random computation $P_0 \xrightarrow[f]{K[0,0]} S_0^r$ and define it to be a base computation. With regard to this base computation, we consider $2^d - 1$ forward differentials Δ_j^K and $2^d - 1$ backward differentials ∇_i^K:

$$0 \xrightarrow[f]{\Delta_j^K} \Delta_j \text{ and } \nabla_i \xleftarrow[f^{-1}]{\nabla_i^K} 0$$

with $(\Delta_0^K, \Delta_0) = (0, 0) = (\nabla_0^K, \nabla_0)$.

This section commences with the presentation of a key theorem related to the construction of bicliques. These bicliques are specifically derived from the properties of independent related-key differentials. We will demonstrate the application and implications of this theorem within a specific sub-cipher configuration $P \underset{f}{\rightarrow} S^r$ derived from the PRINCEv2 lightweight block cipher.

Theorem 3.0.1 *If the trails of both the differentials* $0 \xrightarrow[f]{\Delta_j^K} \Delta_j$ *and* $\nabla_i \xleftarrow[f^{-1}]{\nabla_i^K} 0$ *with* $(\Delta_0^K, \Delta_0) = (0,0) = (\nabla_0^K, \nabla_0)$ *are independent, i.e., they do not share any common active nonlinear components such as S-boxes in* PRINCEv2 *then the following differential trails are ensured:*

$$\nabla_i \xrightarrow[f]{\nabla_i^K \oplus \Delta_j^K} \Delta_j \colon \forall i, j \in \{0, 1, \ldots, 2^d - 1\}.$$

Corollary 3.0.1 *Now by defining* $P_i = P_0 \oplus \nabla_i$, $K[i,j] = K[0,0] \oplus \nabla_i^K \oplus \Delta_j^K$, *and* $S_j^r = S_0^r \oplus \Delta_j$ *using Theorem 3.0.1, one can obtain the following biclique:*

$$P_i \xrightarrow[f]{K[i,j]} S_j^r \colon \forall i, j \in \{0, 1, \ldots, 2^d - 1\}.$$

Theorem 3.0.2 *Let* $C_{Biclique}$ *be the computational complexity due to the biclique. Computational complexity of a biclique of length* r *of size* $2^d \times 2^d$ *in the sub-cipher* f *of* PRINCEv2 *is*

$$\left(1 \times \frac{r}{12}\right) + \left[\left(2^d - 1\right) \times \frac{r}{12}\right] + \left[\left(2^d - 1\right) \times \frac{r}{12}\right].$$

Corollary 3.0.2 *Let* x *and* y *be the number of active S-boxes in* PRINCEv2 *during non-trivial forward and backward differential trails, respectively. Then, the exact computational complexity can be derived from Theorem 3.0.2:*

$$\left(1 \times \frac{r}{12}\right) + \left[\left(2^d - 1\right) \times \frac{x}{192}\right] + \left[\left(2^d - 1\right) \times \frac{y}{192}\right].$$

3.0.2 Overview of the Attack Process

In the biclique cryptanalysis of PRINCEv2, the key space is divided into 2^{120} groups, with each group containing 2^8 keys. For a specific base key (i.e., a fixed key group), the attacker efficiently searches through all 2^8 keys within the group, employing a smart strategy to minimize the complexity of an exhaustive search for identifying the exact master key.

Key Testing Part

In this section, we present a cryptanalytic technique from the attacker's perspective. We first assume that the base key is fixed. Then, we describe how an attacker can identify the correct key from a key group of cardinality 2^8. This process is divided into two main steps:

- The first step is generating S_j^2 using biclique construction.
- The second step involves applying matching with precomputation from S_j^2 to C_i in order to retrieve the correct key either by MITM or SIM. The two steps are described in detail below.

First Step [Biclique Construction Phase]: In the biclique construction phase, the following steps are to be performed:

- Start with plaintext $P_0 = 0$.
- Compute the base computation $P_0 \xrightarrow[f]{K[0,0]} S_0^2$ using the subfunction f under a fixed base key $K[0,0]$.
- Introduce differences in Δ_j^K where $j \in \{0, 1, \ldots, 2^4 - 1\}$ and compute

$$P_0 \xrightarrow[f]{K[0,0] \oplus \Delta_j^K = K[0,j]} S_0^2 \oplus \Delta_j = S_j^2.$$

- Also introduce differences in ∇_i^K where $i \in \{0, 1, \ldots, 2^4 - 1\}$ and compute

$$P_i = P_0 \oplus \nabla_i \xleftarrow[f^{-1}]{K[0,0] \oplus \nabla_i^K = K[i,0]} S_0^2.$$

Since a biclique is established over the first two rounds of PRINCEv2, for any random key $k_{random} = K[i, j]$, one can compute the plaintext P_i and corresponding state after the second round S_j^2.

Second Step [Matching with Precomputation Phase]: In the matching part with the help of precomputation, the attacker can perform the following steps:

- First compute C_i from the plaintext P_i using the oracle. The goal is to match $S_j^2 \xrightarrow[g]{K[i,j]} S_m$ and $S_m \xleftarrow[h^{-1}]{K[i,j]} C_i$ while minimizing the number of active S-box computations to significantly reduce the complexity of exhaustive search.
- From the precomputation $S_j^2 \xrightarrow[g]{K[0,j]} S_m$, recompute only 10 S-boxes in the part $S_j^2 \xrightarrow[g]{K[i,j]} S_m$ and match at the nibble $S_m[12]$.
- Similarly also recompute only 25 S-boxes in the part $S_m \xleftarrow[h^{-1}]{K[i,j]} C_i$ due to the precomputation $S_m \xleftarrow[h^{-1}]{K[i,0]} C_i$ and match with previous one at the nibble $S_m[12]$.

3.1 Significant Results

We propose biclique based attacks on full and reduced round version of PRINCEv2. The details of the attack are as follows:

– A biclique structure of length 2 is constructed in the plaintext side and utilizing precomputation and recomputation techniques the first attack is mounted with a time complexity of $2^{125.95}$ while data complexity and memory complexity are respectively confined to 2^{16} and 2^4.

– Next we inspect PRINCEv2 under a Sieve-in-the-middle (SIM) based biclique approach. The same biclique structure described above is exploited but the subsequent MITM is superseded with a SIM based crypt-analytical procedure. This further reduces the time complexity to $2^{125.66}$ while data complexity and memory complexity are respectively confined to 2^{16} and 2^{20}.

– For a reduced 8-round variant of PRINCEv2, the time complexity of the attack methodology is $2^{124.44}$ with significantly lower data and memory complexities, viz. 2^{16} and 2^4, respectively.

This Chapter is based on the published paper:

- **Himadry Sekhar Roy**, Prakash Dey, Santanu Sarkar, Avishek Adhikari, and Sucheta Chakrabarti (https://doi.org/10.1016/j.dam.2025.10.013, **Discrete Applied Mathematics**).

Chapter 4

Cryptanalysis of Full Round FUTURE with Multiple Biclique Structures

FUTURE, recently proposed in AFRICACRYPT 2022, is a lightweight Substitution Permutation Network (SPN) based block cipher with 10 rounds. In this Chapter, we propose two attack algorithms on FUTURE, both based on biclique structure. The first algorithm uses a biclique structure that is constructed at the ciphertext side between the output state of round 10 (i.e., the ciphertext) and the output state of round 6. The time and memory complexities of the attack are $2^{125.8875}$ and 2^{32}, respectively, while the maximum data complexity is upper bounded by 2^{48}. We further improve upon the attack complexity by introducing one more biclique between the output state of round 6 and the output state of round 2. In the improved attack algorithm, by targeted partitioning and sub-partitioning of the keyspace and by using the biclique structures, a valid key, plaintext, and ciphertext triplet can be obtained for the cipher with time and memory complexities $2^{125.5365}$ and 2^{32}, respectively, while the maximum data complexity is upper bounded by 2^{48}.

4.1 Proposed Biclique Structures in FUTURE

Let P and K respectively be the 64-bit input plaintext and 128-bit secret key. Let C be the 64-bit corresponding ciphertext produced by the cipher i.e., $C = \text{FUTURE}(P, K)$. Let S^r be the output state of the cipher after round r for $r = 1, 2, \ldots, 9$. For the sake of easiness of explanation, we denote C by S^{10}.

We redefine the cipher as the composition of four sub-routines e, f, g and h as FUTURE $= h \circ g \circ f \circ e$ where e, f, g and h are respectively the first 2-round, next 2 round, next 2 round and the remaining 4 rounds of FUTURE. Note that, S^2, S^4, S^6, S^{10} are the respective outputs of e, $f \circ e$, $g \circ f \circ e$ and $h \circ g \circ f \circ e$. In this Chapter we construct biclique structures in each of the following computational parts: $S^6 \underset{h}{\rightarrow} S^{10}$ and $S^2 \underset{g \circ f}{\longrightarrow} S^6$. Let us now formally define a biclique of size $2^m \times 2^n$ corresponding to a computational

part u of the cipher.

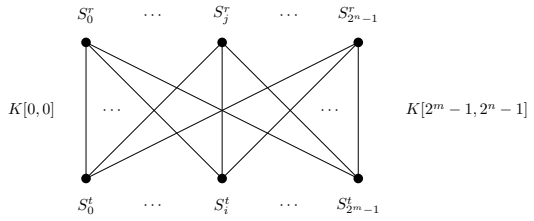

Figure 4.1: A biclique of Size $2^m \times 2^n$ with $S_j^r \xrightarrow[u]{K[i,j]} S_i^t$.

Definition 4.1.1 (Biclique of size $2^m \times 2^n$) *Let $S_0^r, S_1^r, \ldots, S_{2^n-1}^r$ be 2^n output states at round r and $S_0^t, S_1^t, \ldots, S_{2^m-1}^t$ be 2^m output states at round $t(> r)$ of the cipher such that $S_j^r \xrightarrow[u]{K[i,j]} S_i^t$ where $k[i,j]$ is a set of 2^{m+n} keys. Then the collection S_j^r, S_i^t and $K[i,j]$ for $i = 0, 1, \ldots, 2^m - 1$ and $j = 0, 1, \ldots, 2^n - 1$ is said to be a biclique of size $2^m \times 2^n$ (See Figure 4.1). The biclique is between the output of round r and round t. When m and n are equal, say d, then the biclique is said to be of dimension d.*

4.1.1 Formation of Biclique from Differentials

Let $K[0,0]$ be a particular key and S_0^t be given. We construct S_0^r from the computation $S_0^r \xleftarrow[u^{-1}]{K[0,0]} S_0^t$. For $i = 0, 1, \ldots, 2^m - 1$ and $j = 0, 1, \ldots, 2^n - 1$, let Δ_i^K, Δ_i and ∇_j^K, ∇_j be differentials such that $S_0^r \xrightarrow[u]{K[0,0]\oplus\Delta_i^K} S_0^t \oplus \Delta_i$ and $S_0^r \oplus \nabla_j \xleftarrow[u^{-1}]{K[0,0]\oplus\nabla_j^K} S_0^t$. Here Δ_0^K, Δ_0 and ∇_0^K, ∇_0 are zero differentials. We denote $k[i,j] = K[0,0] \oplus \Delta_i^K \oplus \nabla_j^K$, $S_j^r = S_0^r \oplus \nabla_j$ and $S_i^t = S_0^t \oplus \Delta_i$. If the trails of Δ_i-differentials do not share active non-linear components (such as active S-boxes) with the trails of ∇_j-differentials [40,41], then we have the biclique $S_j^r \xrightarrow[u]{K[i,j]} S_i^t$ of size $2^m \times 2^n$. In this case $K[0,0]$ is said to be the base key and $S_0^r \xleftarrow[u^{-1}]{K[0,0]} S_0^t$ is said to be the base computation for the collection $K[i,j]$. The biclique is constructed from S_0^t, $K[0,0]$, using the differentials $\Delta^K = \{\Delta_i^K\}_{i=0}^{2^m-1}$, $\nabla^K = \{\nabla_j^K\}_{j=0}^{2^n-1}$, $\Delta = \{\Delta_i\}_{i=0}^{2^m-1}$ and $\nabla = \{\nabla_j\}_{j=0}^{2^n-1}$. Once such biclique structure is identified and the precomputation $S_0^r \xleftarrow[u^{-1}]{K[0,0]} S_0^t$ is done, the biclique attack proceeds to recompute $S_j^r \xrightarrow[u]{K[i,j]} S_i^t$ by skipping some S-boxes and thus reducing time complexity. Note that the biclique structure internally depends only on u and the differentials Δ^K, ∇^K, Δ and ∇ whereas S_0^t and $K[0,0]$ can be thought of as input parameters. In this case S_j^r for $j = 0, \ldots, 2^n - 1$ and S_i^t for $i = 1, \ldots, 2^m - 1$ are treated as the output of the biclique. With such a background in mind we agree to denote the biclique structure simply by $Biclique\langle u, \Delta^K, \nabla^K, \Delta, \nabla\rangle$.

As stated earlier, we construct two bicliques corresponding to each of the computational parts $S^6 \xrightarrow{h} S^{10}$ and $S^2 \xrightarrow{g\circ f} S^6$ respectively refer them as the first and second biclique. The first biclique is of dimension 32 while the second biclique is of sizes 1×2^{12}.

4.2 Proposed Single Biclique Analysis of FUTURE

We now construct a biclique in the part $S^6 \xrightarrow{h} S^{10}$ by setting $J = J_0 = \{20, 21, \ldots, 51, 64, 65, \ldots, 75, 88, 89, \ldots, 95, 108, 109, \ldots, 119\}$, $\delta^K = \delta^{(K,0)} = \{20, 21, \ldots, 51\}$ and $\epsilon^K = \epsilon^{(K,0)} = \{64, 65, \ldots, 75, 88, \ldots, 95, 108, \ldots, 119\}$ and obtain $Partition\langle B^0, J_0, \delta^{(K,0)}, \epsilon^{(K,0)}\rangle$ as a partition of the Keyspace B^0. It is evident that $Biclique\langle h, \Delta^{(K,0)}, \nabla^{(K,0)}, \Delta^0, \nabla^0\rangle$ indeed a biclique in the part $S^6 \xrightarrow{h} S^{10}$.

4.2.1 Biclique Analysis

Let $k_{random} \in B^0 = \{0,1\}^{128}$ be a random key from the keyspace. Then $k_{random} \in B^1_{K[0,0]} = Span\langle B^0, K[0,0], \delta^{(K,0)}, \epsilon^{(K,0)}\rangle$ for some base key $K[0,0] \in Base\langle J_0/B^0\rangle$.
Let $S_0^{10} \in \{0,1\}^{64}$ be a chosen ciphertext. The first biclique $Biclique\langle h, \Delta^{(K,0)}, \nabla^{(K,0)}, \Delta^0, \nabla^0\rangle$ is constructed from S_0^{10}, $K[0,0]$, using the differentials $\Delta^{(K,0)} = \{\Delta_i^{(K,0)}\}_{i=0}^{2^{32}-1}$, $\nabla^{(K,0)} = \{\nabla_j^{(K,0)}\}_{j=0}^{2^{32}-1}$, $\Delta^0 = \{\Delta_i^0\}_{i=0}^{2^{32}-1}$ and $\nabla^0 = \{\nabla_j^0\}_{j=0}^{2^{32}-1}$. Thus S_j^6 for $j = 0, \ldots, 2^{32}-1$ and S_i^{10} for $i = 1, \ldots, 2^{32}-1$ are obtained as the output of the first biclique. Thus $k_{random} = K[i,j] = K[0,0] \oplus \Delta_i^{(K,0)} \oplus \nabla_j^{(K,0)}$ for some $\Delta_i^{(K,0)} \in \left[\delta^{(K,0)}\right]$ and $\nabla_j^{(K,0)} \in \left[\epsilon^{(K,0)}\right]$ and $S_j^6 = S_0^6 \oplus \nabla_j^0 \xleftarrow[h^{-1}]{K[i,j]} S_i^{10} = S_0^{10} \oplus \Delta_i^0$. In this case, $S_0^6 \xleftarrow[h^{-1}]{K[0,0]} S_0^{10}$ is the precomputation part and $S_j^6 \xleftarrow[h^{-1}]{k[0,j]} S_0^{10}$ and $S_0^6 \xrightarrow[h]{k[i,0]} S_i^{10}$ where $i, j \in \{1, 2, \ldots, 2^{32}-1\}$ are the recomputation part. It follows that for all i and j only 56 S-boxes need to be calculated during the recomputation phase.

4.2.2 Matching with Precomputation

From decryption Oracle, we obtain the plaintext P_i corresponding to the ciphertext S_i^{10} under the unknown key.

In the precomputation part, we now compute $P_i \xrightarrow[f \circ e]{k[i,0]} S_i^4$ and $S_j^4 \xleftarrow[g^{-1}]{k[0,j]} S_j^6$ and store the computation. From this stored computation, we recompute $P_i \xrightarrow[f \circ e]{K[i,j]} S_i^4$ and $S_j^4 \xleftarrow[g^{-1}]{K[i,j]} S_j^6$ with reduced number of S-box evaluation. We denote the 12^{th} cell value of the state S_i^4 i.e., $S_i^4[12]$ by $\overrightarrow{v}_i$ and S_j^4 i.e., $S_j^4[12]$ by $\overleftarrow{v}_j$. If $\overrightarrow{v}_i \neq \overleftarrow{v}_j$ then $K[i,j]$ is not the key. If $\overrightarrow{v}_i = \overleftarrow{v}_j$ then $K[i,j]$ i.e., k_{random} is a possible candidate with probability $\frac{1}{2^4}$ for the computation $P_i \xrightarrow[FUTURE]{\text{Unknown Key}} S_i^{10}$.

4.2.3 Complexity of the Attack

Let C_{Full} be the computation complexity of the attack. The complexity of the attack consists of five parts as described below:

$$C_{Full} = 2^{64} \times [C_{FirstBiclique} + C_{Precomputation} + C_{Recomputation}] + C_{FalsePos}$$

4.3 Significant Results

We propose two biclique attacks on full-round FUTURE.

- The first builds a single biclique from the ciphertext back to round-6 output, recovering the key in $2^{125.8875}$ time, 2^{32} memory, and $\leq 2^{48}$ data.
- The second adds a second biclique from round-6 to round-2, then uses targeted key-space partitioning and sub-partitioning. It yields a valid (key, plaintext, ciphertext) triplet in $2^{125.5365}$ time with identical 2^{32} memory and $\leq 2^{48}$ data bounds.

This Chapter is based on the published paper:

- **Himadry Sekhar Roy**, Prakash Dey, Sandip Kumar Mondal, and Avishek Adhikari (`https://doi.org/10.1007/s12083-023-01600-y`, **Peer-to-Peer Networking and Applications**).

Chapter 5

Enhanced Biclique and MITM Based Key Recovery Attacks on Full and Reduced Round AES-128

This Chapter presents an advanced biclique-based technique applicable to both AES-128 and its reduced-round versions, achieving time complexity improvements with data complexities lower than searching the entire codebook. At the outset of the attack, a biclique of size $2^{40} \times 2^{16}$ with length 3 is initially identified from the ciphertext perspective; however, improved complexity is achieved using a biclique of size $2^{24} \times 2^{16}$. In particular, the method achieves a reduction in the computational complexity for a comprehensive cryptanalysis of AES-128 to $2^{125.91}$, alongside a storage requirement of 2^{24}. By constructing a biclique of size $2^{24} \times 2^{16}$ between rounds 8 and 10, and integrating the sieve-in-the-middle (SIM) cryptanalysis technique, the attack's efficiency is further enhanced, achieving a time complexity of $2^{125.74}$ with a memory complexity of 2^{64}.

Additionally, the novel approach employing both-side storage at the matching point in biclique with improved SIM attack further reduces time complexity to $2^{125.60}$, maintaining the same memory and data complexities. The technique surpasses previous records, notably reducing the time complexity by 25% and 17% compared to the biclique attack without SIM and with SIM respectively by Tao *et al.* (ACISP 2015). For 8-round AES-128, the attack achieves time complexities of $2^{124.02}$ (without SIM) and $2^{123.05}$ (with SIM). Furthermore, the proposed attack employs a new technique to achieve a time complexity of $2^{122.29}$, representing an 84% improvement over the previous best attack by Bogdanov *et al.* at ASIACRYPT 2011. Additionally, we introduce the first biclique-based cryptanalysis for 7-round AES-128, achieving a time complexity of $2^{120.63}$ and a memory complexity of 2^{64}, all under a data complexity bounded by 2^{104}.

Moreover, the MITM attack on 7-round AES-128, utilizing a known single plaintext-ciphertext pair, further refines recent advancements by reducing both data and memory requirements. It achieves time complexity of $2^{125.68}$ and a memory complexity of 2^{32}, and further improves the time complexity to $2^{125.10}$ with memory complexity of 2^{64}.

5.1 Overall Attack Procedure

We outline the procedure for mounting a biclique attack on AES-128. The keyspace $\mathcal{K}_8$ is divided into 2^{88} groups, each containing 2^{40} keys. Key recovery involves two stages: biclique construction and matching with precomputed data.

Finally, retrieve the secret key with probability 1.

Note that we use $P[i_1, i_2, i_3, j_1, j_2]$ instead of $P[i_1, i_2, i_3, 0, 0]$ throughout the biclique attack for verifying the key $K[i_1, i_2, i_3, j_1, j_2]$, where $i_1, i_2, i_3, j_1, j_2 \in \{0, 1, \ldots, 2^8 - 1\}$. For a fixed i_1, i_2, i_3, $P[i_1, i_2, i_3, j_1, j_2] = P[i_1, i_2, i_3, 0, 0]$ is used for verifying the keys $\{K[i_1, i_2, i_3, j_1, j_2] : \ j_1, j_2 \in \{0, 1, \ldots, 2^8 - 1\}\}$. The same applies to S^{10}. In the case of $S^7[0, 0, 0, j_1, j_2]$, for a fixed j_1, j_2, $S^7[i_1, i_2, i_3, j_1, j_2] = S^7[0, 0, 0, j_1, j_2]$ is used for verifying the keys $\{K[i_1, i_2, i_3, j_1, j_2] : \ i_1, i_2, i_3 \in \{0, 1, \ldots, 2^8 - 1\}\}$.

5.2 Significant Results

The actual contributions of the chapter are as follows:

- For AES-128, the approach demonstrates a time complexity of approximately $2^{125.91}$ and a memory complexity of 2^{24} , with a data complexity capped at 2^{104}. By constructing a biclique of size $2^{24} \times 2^{16}$ between rounds 8 and 10 and incorporating the SIM technique during the matching process, we further reduced the time complexity to $2^{125.74}$ while increasing memory usage to 2^{64}. Additionally, by employing improved SIM technique in conjunction with biclique structures, we achieved a further reduction in time complexity to $2^{125.60}$.

- The work also extends to reduced-round versions of AES-128. For 8-round AES-128, we achieved time complexities of $2^{124.02}$ and $2^{123.05}$ with memory complexities of 2^{24} and 2^{64}, respectively. Furthermore, by employing improved SIM technique, we improved the time complexity to $2^{122.29}$. For 7-round AES-128, the attack achieved the best time complexity of $2^{120.63}$ with a memory complexity of 2^{64} and a maximum data complexity bounded by 2^{104}.

- Additionally, we conducted an MITM attack on 7-round AES-128 using a single known plaintext-ciphertext pair. Leveraging the differential properties of the AES state, the attack achieved time complexities of $2^{125.68}$ and $2^{125.10}$, with memory complexities of 2^{32} and 2^{64}, respectively.

This work establishes new benchmarks in biclique-based cryptanalysis for AES-128 and its reduced-round versions, including an MITM attack on 7-round AES-128, enhancing both efficiency and practicality. This Chapter is based on the communicated paper:

- "**Enhanced Biclique and MITM Based Key Recovery Attacks on Full and Reduced Round AES-128**" by **Himadry Sekhar Roy**, Prakash Dey, Sandip Kumar Mondal, Santanu Sarkar, and Avishek Adhikari.

Chapter 6

Low-Depth Quantum Implementation and Quantum Biclique Cryptanalysis of FUTURE

This chapter presents the first formalization of the Quantum Biclique Attack (QBA), a novel framework that extends classical biclique-based cryptanalysis to the quantum domain. By using Quantum Amplitude Amplification (QAA), the proposed method repurposes the classical biclique framework to function within a quantum computational model, thereby establishing a significant theoretical bridge between classical and quantum cryptanalytic techniques. To demonstrate the efficacy of the QBA, we apply the attack to the lightweight block cipher `FUTURE`. Our implementation requires fewer block encryptions per iteration and achieves a reduced T-DW and DW cost of 1.8940×2^{79} and 1.0330×2^{83} respectively. Furthermore, we improve our work with T-DW and DW costing 1.5003×2^{79} and 1.6264×2^{82} respectively. Additionally, we design and analyze the first shallow-depth quantum circuit for `FUTURE`. Utilizing Selinger's Toffoli decomposition, our circuit achieves a depth of 373, a T-depth of 40, and a T-DW cost of $10,240$, representing a substantial optimization in resource requirements compared to the existing decomposition methodologies.

6.1 Quantum Biclique Attack: a General Concept

In this section, we introduce the concept of quantum biclique attack on an arbitrary block cipher E of round r. Let S^i be the output state of E after the round i. For the attack, the cipher E is split into three subroutines f, g, and h such that:

$$\begin{cases} f = E[0, r_2] & \text{(initial part)} \\ g = E[r_2 + 1 : (r - r_1)] & \text{(middle part)} \\ h = E[(r - r_1) : r] & \text{(final part, where biclique is constructed.)} \end{cases}$$

For this, we are required to find an independent classical biclique structure of the block cipher in question, and then implement it in the quantum circuit. Let we consider a classical existence of biclique of size $2^{d_1} \times 2^{d_2}$ in the part $S_j^{r-r_1} \xrightarrow[E[(r-r_1):r]]{K[i,j]} S_i^r$ (or $S_j^{r-r_1} \xrightarrow[h]{K[i,j]} S_i^r$). The main strategy for the quantum implementation is to use entanglement. We exploit the structure of the quantum implementation of ciphers to generate base keys entangled to the key differences as well as their respective states. Note that we will use the notation for gates. In general, for each base key $K[0,0]$, Δ_i^K, ∇_i^K, and the generated states S_i^r and $S_j^{r-r_1}$, we have

$$\frac{1}{2^{\frac{k-d_2}{2}}} \sum_{\substack{K[0,0] \\ \in \mathcal{K}_{\text{base}}}} \sum_{\Delta_i^K=0}^{2^{d_1}-1} |K[0,0]\rangle \otimes \left|\Delta_i^K\right\rangle \otimes |S_i^r\rangle \text{ and } \frac{1}{2^{\frac{k-d_1}{2}}} \sum_{\substack{K[0,0] \\ \in \mathcal{K}_{\text{base}}}} \sum_{\nabla_j^K=0}^{2^{d_2}-1} |K[0,0]\rangle \otimes \left|\nabla_j^K\right\rangle \otimes \left|S_j^{r-r_1}\right\rangle$$

are entangled states.

6.2 Significant Results

The primary objectives attained in this chapter are as follows:

- **Propose Quantum Biclique Framework:** In this chapter, we repurpose the classical biclique framework to operate within a quantum computational model using Quantum Amplitude Amplification (QAA). Moreover, this is effectively bridging theoretical connection between classical and quantum cryptanalysis techniques.
- **Cost of the Low-depth Quantum Circuit of `FUTURE`:** We formally design and analyze the first shallow-depth quantum circuit for the lightweight block cipher `FUTURE`, which optimizes gate depth and quantum resource requirements compared to existing Grover-based approaches. Using Selinger's Toffoli decomposition, we achieve a circuit depth of 373, T-depth of 40 and T-DW cost 10240 which is significantly low as compared to other existing decompositions.
- **Cost of Quantum Biclique Attack (QBA) on `FUTURE`:** Furthermore, we implement and quantify the cost of both the quantum circuit for `FUTURE` and the corresponding QBA, providing concrete depth, width, and T gate complexity estimates. Our proposed attack requires fewer block encryptions per iteration and yields a reduced T-DW and DW cost of 1.8940×2^{79} and 1.0330×2^{83} respectively, demonstrating tangible efficiency gains in both computation and resource scaling. Further, we improve our work with T-DW and DW costing 1.5003×2^{79} and 1.6264×2^{82} respectively.

This chapter is based on the communicated article titled, "**Low-Depth Quantum Implementation and Quantum Biclique Cryptanalysis of `FUTURE`**" by **Himadry Sekhar Roy**, Rajeet Ghosh, Aniket Basak, Santanu Sarkar, and Avishek Adhikari.

Chapter 7

Summary and Future Directions

The initial part of this chapter provides a brief summary of the findings and contributions of each chapter within this thesis. The concluding section delineates the unresolved issues and open problems that warrant further investigation.

Chapter 1 provided an overview of the historical evolution and foundational aspects of cryptology, alongside the advancements leading to contemporary cryptography. It also presented the current state of modern cryptography and cryptanalysis. Furthermore, it outlined the framework for the research conducted in this thesis and concluded with a description of the organization the thesis.

Chapter 2 provided an in-depth exploration of symmetric key cryptographic primitives. Additionally, it provided a comprehensive and current summary of research findings related to the cryptanalysis of these symmetric key cryptographic primitives. The chapter concluded with a detailed overview of several notable block ciphers.

Chapter 3 introduced a cryptanalysis of `PRINCEv2` using biclique-based techniques, focusing on both full-round and reduced-round variants of the cipher. Our analysis reveals that despite enhancements in `PRINCEv2`'s design, including a revised key schedule in the Remark section and keyed middle layer, the time complexity of biclique cryptanalysis increased. Our findings revisit the security assumptions made during `PRINCEv2`'s development.

Chapter 4 demonstrated two attack algorithms on `FUTURE`, both based on the biclique structure. The first algorithm has a time complexity of $2^{125.8875}$ and a memory complexity of 2^{32}, with a maximum data complexity of 2^{48}. The second attack algorithm demonstrated the construction of a valid key, plaintext, and ciphertext triplet for the cipher with a time complexity of $2^{125.5365}$ and a memory complexity of 2^{32}, also with a maximum data complexity of 2^{48}. Our proposed second attack algorithm is novel in that it employs multiple biclique structures instead of a single biclique structure, a method used in classic biclique attacks.

Chapter 5 presented an improved biclique-based cryptanalytic technique targeting full-round `AES-128`, utilizing enhanced biclique structures to achieve significant improvements in attack efficiency. Our work also extends to reduced-round versions of `AES-128`. Additionally, we conducted an MITM attack on 7-round `AES-128` using a single known plaintext-

ciphertext pair. Future research could explore the applicability of these techniques to other block ciphers and investigate potential optimizations for the biclique structures and MITM technique to further reduce computational complexities.

Chapter 6 introduced an innovative Quantum Biclique Attack (QBA) on `FUTURE` along with a low-cost quantum implementation of the block cipher `FUTURE`. The classical biclique attack was adapted to the quantum setting through the application of Quantum Amplitude Amplification (QAA) to effectively address non-uniform entangled states. This adaptation successfully forges a direct theoretical and practical bridge between classical and quantum cryptanalysis methodologies.

The results of this thesis explore the symmetric key primitives through the systematic development and refinement of biclique-based key-recovery techniques, with a dual focus on exposing structural vulnerabilities in modern block ciphers and pushing the efficiency frontier of single-key attacks on the block ciphers.

Future Directions

The section suggests some relevant unresolved issues and open problems for future research:

- To date, there is no known practical key recovery attack against full-round `AES`. While several key recovery attacks on reduced-round `AES` variants exhibit practical time, data, and memory complexities, the only existing attacks on the full-round cipher are theoretical, such as the biclique-based cryptanalytic technique.

- One of the most compelling research problems in the domain of biclique-based cryptanalysis is exploring further improvements in biclique-based techniques for faster key recovery on full-round ciphers like `PRINCEv2`, `AES`, and `FUTURE`.

- One of the most exciting areas of research is the development of new cryptanalysis techniques or the combination of multiple methods. One particularly interesting open problem is exploring the feasibility of combining biclique cryptanalysis with other cryptanalytic techniques.

- In case of quantum cryptography, one of the most exciting research directions is exploring further improvements in the quantum-circuit implementation of `AES`.

- Popular research problems based on Toffoli-gate are exploring further depth reduction in Toffoli-gate circuits available in the literature.

List of Articles Related to the Thesis

Published articles related to the thesis:

1 **Himadry Sekhar Roy**, Prakash Dey, Santanu Sarkar, Avishek Adhikari, and Sucheta Chakrabarti, "**Cryptanalysis of PRINCEv2 using Biclique Structures**", (https://doi.org/10.1016/j.dam.2025.10.013, **Discrete Applied Mathematics**).

2 **Himadry Sekhar Roy**, Prakash Dey, Sandip Kumar Mondal, and Avishek Adhikari, "**Cryptanalysis of Full Round FUTURE with Multiple Biclique Structures**", (https://doi.org/10.1007/s12083-023-01600-y, **Peer-to-Peer Networking and Applications**).

Communicated articles related to the thesis:

3 "**Enhanced Biclique and MITM Based Key Recovery Attacks on Full and Reduced Round AES-128**" by **Himadry Sekhar Roy**, Prakash Dey, Sandip Kumar Mondal, Santanu Sarkar, and Avishek Adhikari.

4 "**Low-Depth Quantum Implementation and Quantum Biclique Cryptanalysis of FUTURE**" by **Himadry Sekhar Roy**, Rajeet Ghosh, Aniket Basak, Santanu Sarkar, and Avishek Adhikari.

Bibliography

[1] Auguste Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, IX(1–2):5–38 (Jan.), 161–191 (Feb.), January 1883.

[2] NIST. National Institute of Standards and Technology: Lightweight Cryptography (LWC) Standardization Project. `https://csrc.nist.gov/Projects/lightweight-cryptography`.

[3] Dirk Fox. Data encryption standard (DES). *Datenschutz und Datensicherheit*, 24(12), 2000.

[4] Whitfield Diffie and Martin E. Hellman. Special feature exhaustive cryptanalysis of the NBS data encryption standard. *Computer*, 10(6):74–84, 1977.

[5] Eli Biham and Adi Shamir. Differential cryptanalysis of DES-like cryptosystems. *Journal of CRYPTOLOGY*, 4:3–72, 1991.

[6] Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.

[7] Joan Daemen and Vincent Rijmen. AES proposal: Rijndael. In *First Advanced Encryption Standard (AES) Conference*, California, USA, Aug 1999.

[8] Joan Daemen and Vincent Rijmen. *The Design of Rijndael: AES - The Advanced Encryption Standard*. Information Security and Cryptography. Springer, 2002.

[9] Martin Hell, Thomas Johansson, and Willi Meier. Grain: a stream cipher for constrained environments. *International Journal of Wireless and Mobile Computing*, 2(1):86–93, 2007.

[10] Steve Babbage and Matthew Dodd. The stream cipher mickey 2.0. Report, ECRYPT Stream Cipher Project, June 2006. Submitted to eSTREAM (Profile 2).

[11] Christophe De Cannière and Bart Preneel. Trivium. In *New Stream Cipher Designs: The eSTREAM Finalists*, volume 4986 of *Lecture Notes in Computer Science*, pages 244–266. Springer, 2008.

[12] Daniel J. Bernstein. The salsa20 family of stream ciphers. In *New Stream Cipher Designs: The eSTREAM Finalists*, volume 4986 of *Lecture Notes in Computer Science*, pages 84–97. Springer, 2008.

[13] Mitsuru Matsui, Itsuo Moriai, and Tetsu Iwata. Acorn: A lightweight authenticated cipher. *Submission to the CAESAR Competition (v3)*, 2016. Lightweight AEAD stream cipher.

[14] Hongjun Wu and Bart Preneel. Aegis: A fast authenticated encryption algorithm (v1.1). Report, CAESAR Competition Submission, 2016. Specification of AEGIS-128, AEGIS-256, and AEGIS-128L.

[15] Jérémy Jean, Ivica Nikolić, and Thomas Peyrin. Deoxys v1.41: A family of lightweight tweakable block ciphers. In *Submission to the CAESAR Competition*, 2016. CAESAR competition submission, final v1.41.

[16] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon: Lightweight authenticated encryption and hashing. *Journal of Cryptographic Engineering*, 9(3):165–180, 2019.

[17] Subhadeep Banik, Sumit Kumar Pandey, Thomas Peyrin, Yu Sasaki, Siang Meng Sim, and Yosuke Todo. GIFT: A small present - towards reaching the limit of lightweight encryption. In Wieland Fischer and Naofumi Homma, editors, *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*, volume 10529 of *Lecture Notes in Computer Science*, pages 321–345. Springer, 2017.

[18] Andrey Bogdanov, Lars R. Knudsen, Gregor Leander, Christof Paar, Axel Poschmann, Matthew J. B. Robshaw, Yannick Seurin, and C. Vikkelsoe. PRESENT: an ultra-lightweight block cipher. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 450–466. Springer, 2007.

[19] Julia Borghoff, Anne Canteaut, Tim Güneysu, Elif Bilge Kavun, Miroslav Knezevic, Lars R. Knudsen, Gregor Leander, Ventzislav Nikov, Christof Paar, Christian Rechberger, Peter Rombouts, Søren S. Thomsen, and Tolga Yalçin. PRINCE - A low-latency block cipher for pervasive computing applications - extended abstract. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Beijing, China, December 2-6, 2012. Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 208–225. Springer, 2012.

[20] Dusan Bozilov, Maria Eichlseder, Miroslav Knezevic, Baptiste Lambin, Gregor Leander, Thorben Moos, Ventzislav Nikov, Shahram Rasoolzadeh, Yosuke Todo, and

Friedrich Wiemer. Princev2 - more security for (almost) no overhead. In Orr Dunkelman, Michael J. Jacobson Jr., and Colin O'Flynn, editors, *Selected Areas in Cryptography - SAC 2020 - 27th International Conference, Halifax, NS, Canada (Virtual Event), October 21-23, 2020, Revised Selected Papers*, volume 12804 of *Lecture Notes in Computer Science*, pages 483–511. Springer, 2020.

[21] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schläffer. Ascon: Lightweight authenticated encryption and hashing. *Journal of Cryptographic Engineering*, 9(3):165–180, 2019. Selected as the NIST lightweight AEAD standard.

[22] Subhadeep Banik, Takanori Isobe, Fukang Liu, Kazuhiko Minematsu, and Kosei Sakamoto. Orthros: A low-latency PRF. IACR Cryptology ePrint Archive, Paper 2021/390; also IACR ToSC 2021, 2021(1):37–77, 2021.

[23] Ravi Anand, Subhadeep Banik, Andrea Caforio, Tatsuya Ishikawa, Takanori Isobe, Fukang Liu, Kazuhiko Minematsu, Mostafizar Rahman, and Kosei Sakamoto. Gleeok: A family of low-latency prfs and its applications to authenticated encryption. *IACR Transactions on Cryptographic Hardware and Embedded Systems (TCHES)*, 2024(2):545–587, 2024.

[24] Himadry Sekhar Roy, Prakash Dey, Santanu Sarkar, Avishek Adhikari, and Sucheta Chakrabarti. Cryptanalysis of princev2 using biclique structures. *Discrete Applied Mathematics*, 379:586–604, 2026.

[25] Himadry Sekhar Roy, Prakash Dey, Sandip Kumar Mondal, and Avishek Adhikari. Cryptanalysis of full round FUTURE with multiple biclique structures. *Peer Peer Netw. Appl.*, 17(1):397–409, 2024.

[26] David Kahn. *The Codebreakers: The comprehensive history of secret communication from ancient times to the internet*. Simon and Schuster, 1996.

[27] Marc Briceno. A pedagogical implementation of a5/1. *http://www. scard. org*, 1995.

[28] Jovan Dj. Golic. Linear statistical weakness of alleged RC4 keystream generator. In Walter Fumy, editor, *Advances in Cryptology - EUROCRYPT '97, International Conference on the Theory and Application of Cryptographic Techniques, Konstanz, Germany, May 11-15, 1997, Proceeding*, volume 1233 of *Lecture Notes in Computer Science*, pages 226–238. Springer, 1997.

[29] Lars R. Knudsen, Willi Meier, Bart Preneel, Vincent Rijmen, and Sven Verdoolaege. Analysis methods for (alleged) RC4. In Kazuo Ohta and Dingyi Pei, editors, *Advances in Cryptology - ASIACRYPT '98, International Conference on the Theory and Applications of Cryptology and Information Security, Beijing, China, October 18-22, 1998, Proceedings*, volume 1514 of *Lecture Notes in Computer Science*, pages 327–341. Springer, 1998.

[30] Scott R. Fluhrer and David A. McGrew. Statistical analysis of the alleged RC4 keystream generator. In Bruce Schneier, editor, *Fast Software Encryption, 7th International Workshop, FSE 2000, New York, NY, USA, April 10-12, 2000, Proceedings*, volume 1978 of *Lecture Notes in Computer Science*, pages 19–30. Springer, 2000.

[31] Scott R. Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the key scheduling algorithm of RC4. In Serge Vaudenay and Amr M. Youssef, editors, *Selected Areas in Cryptography, 8th Annual International Workshop, SAC 2001 Toronto, Ontario, Canada, August 16-17, 2001, Revised Papers*, volume 2259 of *Lecture Notes in Computer Science*, pages 1–24. Springer, 2001.

[32] Itsik Mantin and Adi Shamir. A practical attack on broadcast RC4. In Mitsuru Matsui, editor, *Fast Software Encryption, 8th International Workshop, FSE 2001 Yokohama, Japan, April 2-4, 2001, Revised Papers*, volume 2355 of *Lecture Notes in Computer Science*, pages 152–164. Springer, 2001.

[33] Eli Biham and Orr Dunkelman. Cryptanalysis of the A5/1 GSM stream cipher. In Bimal K. Roy and Eiji Okamoto, editors, *Progress in Cryptology - INDOCRYPT 2000, First International Conference in Cryptology in India, Calcutta, India, December 10-13, 2000, Proceedings*, volume 1977 of *Lecture Notes in Computer Science*, pages 43–51. Springer, 2000.

[34] Alex Biryukov, Adi Shamir, and David A. Wagner. Real time cryptanalysis of A5/1 on a PC. In Bruce Schneier, editor, *Fast Software Encryption, 7th International Workshop, FSE 2000, New York, NY, USA, April 10-12, 2000, Proceedings*, volume 1978 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2000.

[35] Tim Beyne, Yu Long Chen, Christoph Dobraunig, and Bart Mennink. Dumbo, jumbo, and delirium: Parallel authenticated encryption for the lightweight circus. *IACR Trans. Symmetric Cryptol.*, 2020(S1):5–30, 2020.

[36] Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, and Ronny Van Keer. Xoodyak, a lightweight cryptographic scheme. *IACR Trans. Symmetric Cryptol.*, 2020(S1):60–87, 2020.

[37] Christoph Dobraunig, Maria Eichlseder, Florian Mendel, and Martin Schlaffer. Rasta: a cipher with low ANDdepth and few ANDs per bit. In *Advances in Cryptology – CRYPTO 2018, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 662–692. Springer, 2018.

[38] Jincheol Ha, Seonggyeom Kim, Wonseok Choi, Jinseok Lee, Sangyub Lee, Dukjae Moon, HyungChul Won, and Dong-Guk Han. Masta: an HE-friendly cipher using modular arithmetic. *IEEE Access*, 8:194741–194751, 2020.

[39] Kishan Chand Gupta, Sumit Kumar Pandey, and Susanta Samanta. Future: A lightweight block cipher using an optimal diffusion matrix. In Lejla Batina and Joan

Daemen, editors, *Progress in Cryptology - AFRICACRYPT 2022*, pages 28–52, Cham, 2022. Springer Nature Switzerland.

[40] Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger. Biclique cryptanalysis of the full AES. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology – ASIACRYPT 2011*, pages 344–371, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[41] David Wagner. The boomerang attack. In *International Workshop on Fast Software Encryption*, pages 156–170. Springer, 1999.